# Guided genome halving: provably optimal solutions provide good insights into the preduplication ancestral genome of Saccharomyces cerevisiae.

Haris Gavranović, Eric Tannier

Pacific Symposium on Biocomputing 15:21-30(2010)

# GUIDED GENOME HALVING: PROVABLY OPTIMAL SOLUTIONS PROVIDE GOOD INSIGHTS INTO THE PREDUPLICATION ANCESTRAL GENOME OF *SACCHAROMYCES CEREVISIAE*

HARIS GAVRANOVIĆ

*Faculty of Natural Sciences, University of Sarajevo*

ERIC TANNIER

*INRIA Rhône-Alpes ; Université de Lyon ; UMR CNRS 5558, Laboratoire de Biométrie et Biologie Evolutive ; Université de Lyon 1 ; F-69622, Villeurbanne*

We present theoretical and practical advances on the Guided Genome Halving problem, a combinatorial optimisation problem which aims at proposing ancestral configurations of extant genomes when one of them has undergone a whole genome duplication. We provide a lower bound on the optimal solution, devise a heuristic algorithm based on subgraph identification, and apply it to yeast gene order data. On some instances, the computation of the bound yields a proof that the obtained solutions are optimal. We analyse a set of optimal solutions, compare them with a manually curated standard ancestor, showing that on yeast data, results coming from different methodologies are largely convergent: the optimal solutions are distant of at most one rearrangement from the reference.

## 1. Motivations

The genomes of extant species underwent different changes in the course of evolution. These changes are studied at various levels: from modifications at a nucleotide-base level to rearrangements of large pieces of DNA, or duplication of the whole genome.

Genome rearrangement phylogeny methods have matured over the past decade[7] and reached the point where their results can be interpreted together with the results of biologists.

At a low level of resolution, the usually considered mutations that alter genomes are reversals, translocations, fusions and fissions of chromosomes, often included in a general abstract operation called Double Cut-and-Join (DCJ), which has the advantage of being computationally easy to handle and of modelling many realistic rearrangements.[16]

Following El-Mabrouk and Sankoff,[5] Zheng *et al*[18] have generalized the genome rearrangement problems by introducing the possibility of considering a whole genome duplication in the genome histories. They provide a heuristic algorithm[19] which is able to propose the organization of a set of genes along the history of yeast genomes.

In the same time, Gordon *et al*[8] chose not to use an automatic method to reconstruct the preduplication ancestor of *Saccharomyces cerevisiae*. Their arguments, among others, were that (i) only a small subset of the genes (less than 20%) can be taken into account, those which have retained two copies after the WGD, and (ii) the optimization problems are computationally complex and the methods are still in development.

In this paper, we report some advances on the theoretical study of the guided genome halving problem, and apply a heuristic algorithm on "double conserved syntenies" of different yeast species. As shown in a previous study,[12] the computation of double syntenies allows to apply the guided halving problem on instances that cover a good ratio of the exant genes (over 95%), and the computation of a lower bound proves that on some instances, the algorithm reaches an optimal solution. It is a solution to the two mentioned objections of Gordon *et al*,[8] and the comparison of the obtained solutions with the manually constructed one shows a good convergence. There were other objections in the paper of Gordon *et al*, as well as in a comment of Sankoff[9] in the same journal, as the possible huge number of optimal solutions, or the inability of the models to account for certain types of rearrangements, like telomeric translocations. This still calls for a theoretical answer (though telomeric translocations are actually taken into account by the DCJ framework) whereas the surprising convergence between all the solutions we find on one instance and the manually reconstructed ancestor shows that some dataset are already accessible.

Pacific Symposium on Biocomputing 15:21-30(2010)

In the next section, we present the mathematical definitions of the genomes and the rearrangements that may alter them. Then in Section 3, we propose a lower bound on the guided halving solutions, based on the so-called "double distance" computation. This bound may be reached, so is able to prove optimality of some solutions. Its usage in a branch and bound algorithm calls for efficient ways to compute it, since the double distance computation is NP-hard for the DCJ distance, and we are only able to compute its exact value for the easiest instances here. In Section 4, we describe the principle of our heuristic algorithm, and compare its efficiency with the state of the art one of Zheng *et al*[19] on some common instances build from yeast genomes, but with a low coverage of the whole genomes. Finally in Section 5, we apply this algorithm on good coverage syntenies on yeast genomes, and provide some information on the history of *Hemiascomycetes*, which we can compare to the ones Gordon *et al*. They are very similar, and give the hope that algorithmic rearrangement studies can provide good insights into genome evolution.

## 2. Genomes and rearrangements

### 2.1. *Genes and genomes*

We use the standard algorithmic definitions of genes and genomes[3,13]*. A *gene* $A$ is an oriented sequence of DNA nucleotides, identified by its *tail* $A^t$ and its *head* $A^h$. Tails and heads are called *extremities* of the gene. An *adjacency* is an unoriented pair of gene extremities. A *genome* $\Pi$ is a set of adjacencies on a set of genes, such that every gene extremity participates in at most one adjacency. In a genome, an adjacency means that two genes are consecutive on the DNA molecule. In a genome $\Pi$, a gene extremity which is not adjacent to another gene extremity is called a *telomere*. A telomere $a$ is also written as an adjacency $a\circ$, where $\circ$ is an abstract symbol not related to a gene, and called a *telomeric adjacency*.

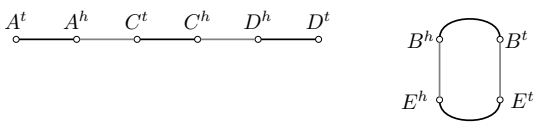
For given genome $\Pi$, the *genome graph* $G_\Pi$ has vertex set the set of all gene extremities, and edge set the union of non telomeric adjacencies and edges $A^tA^h$, called *gene edges*, for every gene $A$. This graph has vertices of degree one or two. Thus, connected components are paths and cycles, and are called the *chromosomes* of $\Pi$. Paths are *linear* chromosomes, whereas cycles are *circular* chromosomes. Telomeres are degree one vertices of the genome graph. A genome with only linear, or only circular, chromosomes is called linear or circular genome, respectively.

A genome can also be represented as a set of strings, by writing the genes for each chromosome in the order in which they appear in the paths and cycles with a bar over the gene if the head of the gene appears before the tail and none if the tail appears before the head (this depends on an arbitrary direction of reading). For each linear chromosome, there are two possible equivalent strings, for two opposite traverses of the path. We have also two oposite circular strings for circular chromosomes.

**Example 2.1.** Let

$$\Pi = \{A^hC^t, C^hD^h, B^hE^h, E^tB^t\}$$

be a genome with five genes $\{A, B, C, D, E\}$. The corresponding genome graph is the following:




and the string representation consists in the linear string $AC\bar{D}$ (or $D\bar{C}\bar{A}$) and the circular string $B\bar{E}$ (or $E\bar{B}$).

*They do not meet the biological definition of genes, but are more precisely "families of homologous sequences", containing from 0 to several genes in each genome.

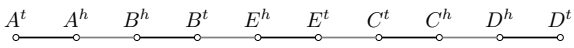
Pacific Symposium on Biocomputing 15:21-30(2010)

A *Double Cut-and-Join* (DCJ) operation (sometimes called 2-break rearrangement) is defined for two adjacencies $pq$ and $rs$ of a genome. Telomeric adjacencies are also considered and any $p, q, r$ or $s$ could be a $\circ$ symbol. Moreover even the adjacency $\circ\circ$ is considered. Thus, the DCJ transforms two adjacencies $pq$ and $rs$ into either $pr$ and $qs$, or $ps$ and $qr$. The DCJ was introduced by Yancopoulos *et al*[16] to encompass all interesting types of genomic rearrangements. Indeed, an inversion, a chromosome fusion or fission, as well as a reciprocal translocation, is a particular case of a DCJ. A non-reciprocal translocation, meaning a chromosome arm is translocated to another chromosome, is also a DCJ. Transpositions and block interchanges can be mimicked by two consecutive DCJs, the intermediate genome containing a circular chromosome.

**Example 2.2.** Consider genome from Example 2.1 and the DCJ that transforms adjacencies $A^hC^t, B^hE^t$ into $A^hB^h, C^tE^t$. The new genome is then

$$\Pi' = \{A^hB^h, E^hB^t, C^tE^t, C^hD^h\}$$

and its genome graph becomes




Given a genome $\Pi$, it is always possible to transform it into another arbitrary genome on the same set of genes applying a sequence of DCJ operations. This leads to the definition of DCJ distance.

**The DCJ Sorting and Distance Problem.** Given two genomes $\Pi$ and $\Psi$ defined on the same set of genes, find a shortest sequence of DCJ operations that transforms $\Pi$ into $\Psi$. The length of such a sequence is called the *DCJ distance* between $\Pi$ and $\Psi$, denoted by $d_{DCJ}(\Pi, \Psi)$.

As the DCJ distance is the main distance we consider here, $d_{DCJ}$ is often abbreviated by $d$. When three genomes are considered, this yields the Median problem:

**The Genome Median Problem.** Given three genomes $\Pi_1$, $\Pi_2$, $\Pi_3$, find a *median* genome $\Pi$ wich minimises $d_{DCJ}(\Pi_1, \Pi)+d_{DCJ}(\Pi_2, \Pi)+d_{DCJ}(\Pi_3, \Pi)$.

The *breakpoint graph* of a set of genomes on the same set of genes is the graph on vertex set the extremities of all genes, and edge set the set of adjacencies of all genomes. If an edge is an adjacency on a genome $\Pi$, it is called a $\Pi$-edge.

Each vertex of the breakpoint graph has degree at most the number of considered genomes. So for two genomes, it is a set of paths and cycles. The DCJ distance is easily computed from the breakpoint graph: for two genomes $\Pi$ and $\Psi$ on $n$ genes, if $C$ is the number of cycles and $P$ the number of paths with an even number of edges (including trivial path with 0 edges),

$$d(\Pi, \Psi) = n - (C + P/2).$$

The DCJ sorting and distance problems thus have a linear time running solution.[3] The genome median problem, however, is NP-hard[13] under the DCJ distance, thought there are good algorithms to solve it.[11,15]

### 2.2. *Duplicated genes and genomes*

A *duplicated gene* is a couple of homologous oriented sequences of DNA nucleotides, identified by two tails $A_1^t$ and $A_2^t$, and two heads $A_1^h$ and $A_2^h$. An *all-duplicates* genome $\Delta$ is a set of adjacencies on a set of duplicated genes, where each gene extremity is contained in at most one adjacency.

For a genome $\Pi$ on a gene set, a *doubled genome* $\Pi\oplus\Pi$ is an all-duplicates genome on the set of duplicated genes from the same gene set such that if $A^xB^y$ $(x, y \in \{t, h\})$ is an (possibly telomeric) adjacency of $\Pi$ ($A^x$ or $B^y$ may be $\circ$), either $A_1^xB_1^y$ and $A_2^xB_2^y$, or $A_1^xB_2^y$ and $A_2^xB_1^y$, are adjacencies in $\Pi \oplus \Pi$.

We note that on a doubled genome one finds two identical copies of each chromosome when we ignore the $1's$ and $2's$ in the names of genes. More precisely, it has two copies of each linear chromosome, and for each

Pacific Symposium on Biocomputing 15:21-30(2010)

circular chromosome, either two circular copies or one circular chromosome containing the two successive copies. For one genome $\Pi$ there is an exponential number of possible doubled genome $\Pi \oplus \Pi$.

The DCJ distance and median problems generalize to the case of duplicated genomes in several ways.

The DCJ distance between two all-duplicates genomes is easily derived from the usual DCJ distance and has a polynomial time computation. But if we ignore the $1's$ and $2's$ in the names of genes, then then it calls for an re-assignement of $1's$ and $2's$ which minimizes the distance, and this problem is NP-complete.

**The double distance problem.** The *double distance* between an ordinary genome $\Pi$ and an all-duplicates genome $\Delta$ is defined as $d(\Pi, \Delta) = min_{\Pi \oplus \Pi} d(\Pi \oplus \Pi, \Delta)$. It is also NP-hard.[13]

**The Genome Halving Problem.** Given an all-duplicates genome $\Delta$ on a set of duplicated genes, find a doubled genome $\Pi \oplus \Pi$ on the same set of genes such that the DCJ distance between $\Delta$ and $\Pi \oplus \Pi$ is minimal. The DCJ distance $\min_{\Pi \oplus \Pi} d(\Pi \oplus \Pi, \Delta)$ (the genome halving score) is denoted by $gh(\Delta)$.

The genome halving problem aims at constructing possible preduplication configuration of genomes which have undergone a whole duplication in the course of their histories. Computationnally, it is a generalization of the DCJ sorting problem (indeed if the two copies of the ancestral doubled genome evolve independently and no rearrangement concerns both, then it is equivalent to simple DCJ sorting). It was solved in the most complicated case where only linear chromosomes are allowed by El-Mabrouk and Sankoff,[5] resulting in a rather complicated algorithm. Alekseyev and Pevzner discuss and solved the same problem on unichromosomal genomes.[1] Recently, solutions with DCJ distance were presented by Warren and Sankoff[14] and Mixtacki.[6]

The solution relies on the *contracted breakpoint graph*[†] of the genome $\Delta$: its vertex set is the set of pairs of homologous extremities of duplicates genes (two extremities $(A_1^h, A_2^h)$ or $(A_1^t, A_2^t)$ form a single vertex). Two vertices are connected by an edge if two extremities are adjacent in $\Delta$. This graph is a set of cycles and paths. If we call $n$ the number of genes (counting one for the two duplicates), $EC$ the number of cycles of even length and $EP$ the number of paths with an even number of edges (including trivial paths with no edges), then Mixtacki[6] proved that

$$gh(\Delta) = n - (EC + \lfloor \frac{EP}{2} \rfloor).$$

This formula yields a linear algorithm to solve the Genome Halving problem. The analysis of this algorithm shows the existence of a large, often exponential, number of optimal solutions. This fact makes it inappropriate for any practical, biologically significant computation. Seoighe and Wolfe[10] noted this extreme non-uniqueness of solution to genome halving problem and propose to use a reference genome, *i.e.* outgroup to reduce this number. Zheng *et al*[18] formalize this approach and propose the first computational method to solve it following with more recent[19] and more efficient method applied to find phylogenetic relationships among yeasts of the *Saccharomyces* complex. Here they also define the Genome Halving problem with two outgroups.

**The Guided Genome Halving Problem.** Given an all-duplicates genome $\Delta$ and an ordinary genome $\Gamma$ defined on the same set of genes, find an ordinary genome $\Pi$ which minimizes

$$ggh(\Delta, \Gamma) = d_{DCJ}(\Delta, \Pi) + d_{DCJ}(\Pi, \Gamma).$$

These problems have different variants depending of the possibility for a genome of having one or several chromosomes, only linear chromosomes or only circular chromosomes or a mix between the two. In what follows we consider the genomes with several, exclusively linear chromosomes and the distance is DCJ distance as defined here. The problem is a generalization of the median problem and is NP-hard as shown by Tannier et al. and by Zheng et al.[13,19]

[†]It is a generalization to linear chromosomes of the "contracted breakpoint graph" defined by Alexseyev and Pevzner,[1] and is the line-graph of the "natural graph" used by Mixtacki.[6]

Pacific Symposium on Biocomputing 15:21-30(2010)

## 3. A lower bound

For the median problem, there is a usual folklore lower bound that is used for very efficient branch and bound approaches.[15] Indeed, for three genomes $\Pi_1$, $\Pi_2$ and $\Pi_3$, and a median genome $\Pi$, it is trivial from the triangle inequality that

$$d_{DCJ}(\Pi_1,\Pi)+d_{DCJ}(\Pi_2,\Pi)+d_{DCJ}(\Pi_3,\Pi)$$
$$\geq \frac{d_{DCJ}(\Pi_1,\Pi_2)+d_{DCJ}(\Pi_1,\Pi_3)+d_{DCJ}(\Pi_3,\Pi_2)}{2}.$$

The Guided Genome Halving problem is a generalization of the median problem (indeed, if from the ancestor the two copies of the doubled genome evolve independenly and no rearrangement mixes the two, then it is equivalent to the median problem). But no such bound exists for this problem. We draw an equivalent, though less trivial and less computationally easy, which helps evaluating solutions.

**Theorem 3.1.** *Given an all-duplicates genome $\Delta$ and an ordinary genome $\Gamma$ defined on the same set of genes,*

$$ggh(\Delta,\Gamma) \geq \frac{d_{DCJ}(\Gamma,\Delta)+gh(\Delta)}{2}.$$

Indeed, it is an easy exercise to show that the double distance verifies the triangle inequality. This yields, for any genome $\Pi$ and doubled genome $\Pi\oplus\Pi$,

$$d(\Gamma\oplus\Gamma,\Delta) \leq d(\Gamma\oplus\Gamma,\Pi\oplus\Pi)+d(\Pi\oplus\Pi,\Delta).$$

Clearly, $\Gamma\oplus\Gamma$ can be chosen so that $d(\Gamma\oplus\Gamma,\Pi\oplus\Pi)\leq 2d(\Gamma,\Pi)$, which gives

$$d(\Gamma\oplus\Gamma,\Delta) \leq 2d(\Gamma,\Pi)+d(\Pi\oplus\Pi,\Delta).$$

Adding $d(\Pi\oplus\Pi,\Delta)$ to both sides, we get

$$d(\Gamma\oplus\Gamma,\Delta)+d(\Pi\oplus\Pi,\Delta) \leq 2(d(\Gamma,\Pi)+d(\Pi\oplus\Pi,\Delta)).$$

And finally, as by definition $gh(\Delta)\leq d(\Pi\oplus\Pi,\Delta)$ and $d(\Gamma,\Delta)\leq d(\Gamma\oplus\Gamma,\Delta)$,

$$\frac{d(\Gamma,\Delta)+gh(\Delta)}{2} \leq d(\Gamma,\Pi)+d(\Pi\oplus\Pi,\Delta).$$

Which, for a genome $\Pi$ such that $ggh(\Gamma,\Delta)=d(\Gamma,\Pi)+d(\Pi\oplus\Pi,\Delta)$, yields the result

The bound may be reached only if the optimal solution of the guided halving problem is also an optimal solution to the halving problem for the all-duplicates genome.

It is based on the computation of the double distance, which is NP-complete.[13] So it is not immediately usable in a branch and bound algorithm as the one for the median problem.[15] A less tight bound may be used by replacing $d_{DCJ}(\Gamma,\Delta)$ by $d_{BP}(\Gamma,\Delta)/2$, where $d_{BP}$ is the double breakpoint distance and is computed with a linear algorithm (see Tannier *et al*[13]). This more tractable bound is less often reached and never allows to prove optimality in our case. For the easiest instances on yeast data (see Section 5), we could compute the DCJ bound exactly and it allows to prove optimality of our Guided Genome Halving solutions given by the algorithm below.

## 4. The Algorithm for Guided Genome Halving

### 4.1. *Contracted breakpoint graphs*

Previous work and experience on solving the problem computationally come from Sankoff's group.[18,19] The first approach[18] was to generate all possible genome halving solution and thereafter choose the subset of solutions that minimize the distance from the outgroup. At the end, the authors develop local improvement

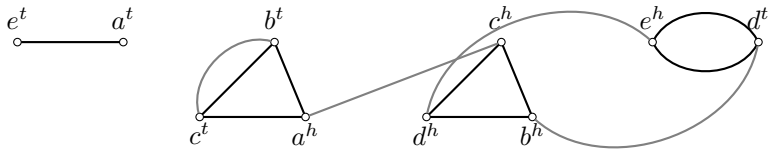
Pacific Symposium on Biocomputing 15:21-30(2010)

heuristic searching in the neighbourhood of optimal halving solutions in order to find better solutions. Due to potentially huge number of halving solutions only numerical results for maize with 34 doubled blocks are reported. For any bigger instance one is obliged to choose heuristically a small subset of halving solutions and proceed with the method, therefore trade off the time and quality of solutions.

The inspiration for our algorithm is the idea from Zheng *et al*[19] where the authors combine information from both the all-duplicates genome and the outgroup early in the process of constructing the ancestor. We will use a small concrete genome with 5 duplicated genes to explain the main idea of the algorithm, and we will use the genome of Example 2.1 as an outgroup.

**Definition 4.1.** Let $\Delta$ be all-duplicates genome and let $\Gamma$ be an ordinary genome defined on the same set of genes. The *contracted breakpoint graph* of $\Delta$ and $\Gamma$ is the graph built on the contracted breakpoint graph of $\Delta$ (which edges are called the red edges), by adding an edge (called blue edge) for each adjacency of $\Gamma$. We note the obtained graph $B(\Delta, \Gamma)$.

Example 3: Let $\Delta = a_1 b_1 \bar{c_1} b_2 \bar{d_1} \bar{e_1} a_2 c_2 \bar{d_2} \bar{e_2}$ be an all-duplicated genome. Let $\Gamma = a\bar{c}bd\bar{e}$ be an ordinary linear unichromosomal genome. The associated contracted breakpoint graph follows (red edges are drawn bold, while blue ones are thin).



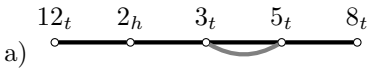

### 4.2. *The algorithm*

We used similar ideas as in Alekseyev and Pevzner[2] or Zhao and Bourque,[17] who find "reliable rearrangements" in a multiple breakpoint graph, as well as in Xu,[15] who searches for "adequate subgraphs" in a multiple breakpoint graph. The two are the dynamic and static versions of the same principle: indentifing some subgraphs in a breakpoint graph (whether it is a multiple or a contracted breakpoint graph changes only the details) for which there is a provably optimal local ancestral arrangement, and thus rearrangements. That is, for some patterns on the graph, it is possible to draw a realiable ancestor, and then to restrict the heuristic principles on the rest of the graph, which we expect much smaller.

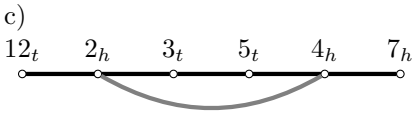
A DCJ operation on genome $\Delta$ is immediately transposable on the contracted breakpoint graph: it consists in deleting two edges (or one edge and one telomere), and join the 4 pending vertices by two other edges. We chose the dynamic approach of Alekseyev and Pevzner[2] or Zhao and Bourque,[17] identifying reliable rearrangements: we start with the contracted breakpoint graph of $\Delta$ and $\Gamma$, and apply DCJ operations in sequence until all red edges are doubled, which means we reached a doubled genome, so a solution to the guided genome halving problem. We detect the following three configurations:

- We apply first all DCJs that directly lead to a red-blue cycle of length two;
- then we choose small sequences of DCJ leading to red-blue cycles of length two, at the condition that it is not destroying any other red-blue two cycle. We can recognize such a sequence in the contracted breakpoint graph: it is drawn from cycles consisting of some red edges and one blue edge, while other blue edges adjacent to cycle are adjacent to red connected components (see Figure 1). The sequence can be shuffled to diversify the final solution.
- finally choose sequences of DCJ leading to red-blue cycles of length four, at the condition that it is not destroying immediately any red-blue 2-cycle. We recognize such a sequence in the contracted breakpoint graph as a cycle consisting of some red edges and two blue edges while other blue edges adjacent to the cycle are adjacent to other red connected components of the contracted breakpoint graph (see Figure 1). The sequence can be shuffled to diversify the final solution.

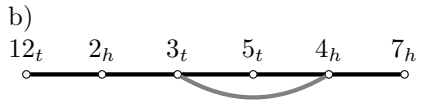
Pacific Symposium on Biocomputing 15:21-30(2010)



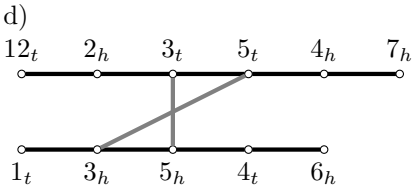

This is a straitforward case where a red edge is parallel to a blue edge. We choose the DCJ that transforms $(2_h, 3_t), (5_t, 8_t)$ into $(5_t, 3_t), (2_h, 8_t)$, constructing a double-red/blue cycle.




In this case, there is a red-blue cycle with adequated parity. A first DCJ connects $3_t$ and $5_t$ while the other connects $2_h$ and $4_h$. Note that any order of these DCJs will construct one double-red/blue cycle.




Here there is a red-blue cycle but its parity does not allow a sequence of DCJs that construct a double-red/blue cycle.




This configuration forces the choice of two DCJs: one that connects $3_t$ and $5_t$ while the other connects $3_h$ and $5_h$. These operations lead to a double-red/blue cycle of size 4 that is solved with only one DCJ.

Fig. 1. The detected DCJ rearrangements. Red edges are drawn black and blue edges are drawn grey.

When no such pattern is found, we apply series of DCJ which all lead to an optimal genome halving ancestor, with a randomized choice.

### 4.3. *Results on Sankoff's instances on yeast genomes*

Zheng *et al*[19] use their Guide Genome Halving Heuristic on several instances built from yeast genes (personnal communication). They choose a pool of genes from the Yeast Gene Order Browser,[4] namely those which are in one exemplar in the non duplicated species and in exactly two exemplars in the duplicated species. So a minority of the genes are covered, make it difficult to compare with the manual reconstructed ancestor of Gordon *et al.*[8] However they provide a good benchmark for comparison with our method, showing that we come to similar performances. We achive slighlty better solutions on the majority of instances. Results are presented on Table 1.

## 5. Results on high coverage yeast data

### 5.1. *Double conserved syntenies*

The instances of the Guided Genome Halving problem were constructed by a double conserved synteny method described in an earlier paper,[12] which roughly consists in, given orthologies between a duplicated genome and a non-duplicated genome (all orthologies are taken from the YGOB[4]) looking for a set $S$ of gene families which have one gene in the first genome, and at least one in the second, and verify

(1) the genes of $S$ are contiguous in $Lk$;
(2) the genes of $S$ form two contiguous segments $A_{Sc}$ and $B_{Sc}$ in $Sc$, of at least 2 genes each;
(3) the two sets $A_{Lk}$ and $B_{Lk}$ of genes of $Lk$ which have homologs respectively in $A_{Sc}$ and $B_{Sc}$ form two intersecting segments in $Lk$;
(4) At least one extremity of $A_{Sc}$ (resp. $B_{Sc}$) is homologous to an extremity of $A_{Lk}$ (resp. $B_{Lk}$);
(5) $S$ is maximal for these properties.

The first two conditions impose the presence of one segment in $Lk$ and two orthologous segments in $Sc$, with a minimum size. It is the basis of the double synteny signal. The presence of at least two genes

Pacific Symposium on Biocomputing 15:21-30(2010)

| Instance Γ - Δ | $2n$ | $d(\Delta, X * X)$ | $d(X, \Gamma)$ | $d(X, \Gamma)$ |
|---|---|---|---|---|
| AG-CG | 538 | 186 | 153 | 144 |
| AG-SC | 1012 | 119 | 188 | 187 |
| KL-CG | 546 | 186 | 147 | 147 |
| KL-SC | 1026 | 122 | 197 | 197 |
| KW-CG | 542 | 188 | 215 | 205 |
| KW-SC | 994 | 121 | 323 | 317 |
| A*-CG | 600 | 199 | 84 | 71 |
| A*-SC | 1062 | 124 | 5 | 8 |
| AG-V | 576 | 61 | 148 | 149 |
| KL-V | 584 | 62 | 157 | 155 |
| KW-V | 582 | 62 | 212 | 210 |
| A*-V | 600 | 62 | 29 | 27 |

Table 1. Comparison of the results obtained in Zheng *et al*[19] and by our algorithm. The first column indicated the couple of compared species. The number of duplicated genes (in these instances they are the synteny blocks) is reported in the second column while in the third on we find the genome halving distance. All the best solutions we found are also genome halving solutions. The forth and the fifth column contain distances from the solution the outgroup as reported by Zheng *et al* and obtained in this study. AG stands for *E. gossypii*, CG for *C. glabrata*, SC for *S. scerevisiae*, KL for *K. lactis*, KW for *K. waltii*, A* for the pre-duplication ancestor of SC and CG, and V for the post-duplication last common ancestor of SC and CG. These are the notations used by Zheng *et al.*

avoids the possible fortuitous presence of one transposed or misannotated gene. The third condition avoids the ambiguous signal of two successive single syntenies. The fourth condition is used to orient the markers.

In this way, we were able to compare every pair of duplicated yeast (2 assembled species) and non-duplicated yeast (5 assembled species). The coverage of the genomes is always above 95%, which allows to reconstruct a large part of the history of the genomes.

### 5.2. *The alternative ancestors*

One first surprising thing already remarked in the earlier study[12] is that the solutions to the Guided Genome Halving on two species (*Saccharomyces cerevisiae* and *Lachancea kluyverii*) and the manually constructed ancestor of Gordon *et al*[8] from 11 species come very close. It is confirmed here, by the examination of several solutions.

First of all, on this instance, our program (as well as the one of Zheng *et al*[19]) finds an optimal solution: indeed, the value of the solution is 140, while the bound gives 139.5. This instance is one of the few for which the bound is tractable without involving deep algorithmics. So it is a good information that no arrangement can be strictly more parsimonious than the ones we find.

Our solutions are sequences of DCJs on the all-duplicates genome. 26 different sequences lead to an optimal solutions, and among the 26 solutions, only two are different. They vary by one reciprocal translocation and are both distant of one telomeric translocation from the solution of Gordon *et al*,[8] which is suboptimal for the number of DCJ (score 141).

The only point in which all three results vary is the position of a small part of Gordon *et al*[8]'s ancestral chromosome 1 (Anc1.1-Anc1.120), which is alternatively fusioned to ancestral chromosome 2 or 6.

### 5.3. *The rearrangements*

Gordon *et al*[8] have also manually infered all the rearrangements from the ancestral genome to *Saccharomyces cerevisiae*. They found in total 144 rearrangements, 73 being inversions, 66 reciprocal translocations and 5 telomeric translocations.

Pacific Symposium on Biocomputing 15:21-30(2010)

We find 115 DCS between our ancestor and the genome of *Saccharomyces cerevisiae*, and 116 between Gordon *et al*'s ancestor. The difference is probably partly due to our definition of Double Conserved Syntenies, which allow local rearrangements to a certain extant. Rearrangement distances are always difficult to compare at different levels of resolution. But interchromosomal rearrangements are comparable, since rearrangements inside the makers should be only inversions.

There is no unique scenario as pointed by Gordon *et al*,[8] and some types of rearrangements are difficult to assess with certainty. But for a pool of 26 scenarios, we found around 20 inversions, 7 couples of DCJs transposing or exchanging the positions of blocks, 70-71 reciprocal translocations, and 7-8 telomeric translocations. The number of reciprocal translocations and telomeric translocations hashas the same order than in the manually reconstructed scenario. Inversions are much less numerous, even if transposition couples of DCJs are counted as 3 inversions. So most inversions seem so be included local rearrangements within the syntenies. No current method can faithfully evaluate this number of local rearrangements which involve double conserved syntenies.

Gordon *et al*[8] only make their analyses on the *Saccharomyces cerevisiae* branch. Here, we are able to reconstruct also the whole distance matrix between all yeast species available in YGOB.[4] The results, in terms of numbers of DCJs, are reported in Table 2. The tendecies in the rearrangement rates on all species follow the "number of blocks" statistic of Gordon *et al*.[8] A multiple study would identify the shared rearrangements, but this is left for a future work.

| | E.gossypii | K.lactis | L.thtolerans | L.waltii* | L.kluyveri | Z.rouxii |
|---|---|---|---|---|---|---|
| S.cerevisiae | 117 + 163 | 116 + 183 | 111 + 67 | 114 + 84 | 115 + 25 | 119 + 118 |
| S.bayanus* | 156 + 164 | 140 + 175 | 168 +109 | 166 + 120 | 176 + 83 | 173 + 144 |
| C.glabrata | 251 + 177 | 258 + 189 | 266 + 108 | 256 + 124 | 266 + 75 | 273 + 136 |
| N.castellii* | 177 + 169 | 166 + 189 | 192 + 106 | 192 + 118 | 199 + 81 | 195 + 146 |
| V.polyspora* | 199 + 173 | 192 + 182 | 220 + 101 | 216 + 117 | 225 + 78 | 215 + 146 |

Table 2. DCJ distances between pairs of genomes. $A+B$ means: $A$ is the distance from the ancestor to the duplicated species, and $B$ is the distance from the ancestor to the non-duplicated species. So on one line, we expect the first number to be approximatively the same (up to a variation on the number of local rearrangements inside the markers, which vary for each comparison). Species which name is followed by an asterisk are not yet assembled, so probably the number of rearrangements is overestimated.

## 6. Conclusion

We presented an algorithm for the Guided Genome Halving problem, as well as a lower bound. The algorithm uses ideas similar to the ones of Zheng *et al*,[19] accompagnied by principles used for the median problem by Xu,[15] or for multiple comparisons by Alekseyev and Pevzner[2] or Zhao and Bourque,[17] which is natural as the guided halving generalizes the median. On some instances coming from yeast order data, the bound gives a proof that the obtained solution is optimal. Comparing a set of optimal solutions with a standard preduplication ancestor of *Saccharomyces cerevisiae*, we obtain two interesting conclusions : the standard arrangement is sub-optimal, at one operation from optimal solutions, and the latter vary only by the position of a single block.

This shows that on yeast data, it seems that the Guided Halving problem provides a very good modelization, perhaps better than on mammalian data, where automatic methods have diverged from standard manual studies for a while.

Future work will concern the efficient computation of the double distance problem, in order to provide a bound which is possible to use within an algorithm for the Guided Genome Halving. Zheng *et al*[19] have also generalized the Guided Halving to instances with two non-duplicated genomes. The data and study from Gordon *et al*[8] also calls for the possibility of reconstructing the full *Saccharomycetes* phylogeny on several duplicated as well as non duplicated species.

Pacific Symposium on Biocomputing 15:21-30(2010)

**Ackownledgements**

We thank Chunfang Zheng for kindly providing her program and the instances on which we could make a benchmark comparison. Thanks to Ken Wolfe for the informations on the unique locus which seems to be displaced in the guided genome halving solutions. This work was supported by the Centre National de la Recherche Scientifique, and by the Agence Nationale de la Recherche (ANR-08-GENM-036-01).

## References

1. Alekseyev MA, Pevzner PA (2008) Colored de Bruijn graphs and the genome halving problem, IEEE/ACM Transactions on Computational Biology and Bioinformatics 4:98-107
2. Alekseyev MA, Pevzner PA (2009) Breakpoint graphs and Ancestral Genome Reconstruction, Genome Research 19(5):943-57
3. Bergeron A. Mixtacki J, Stove J (2006) A unifying view of genome rearrangments, Proceedings of WABI'06, Springer, Lecture Notes in Computer Science 4175:163-173
4. Byrne KP, Wolfe KH (2005) The Yeast Gene Order Browser: combining curated homology and syntenic context reveals gene fate in polyploid species. Genome Research 15(10):1456-61
5. El-Mabrouk N, Sankoff D (2003) The reconstruction of doubled genomes, SIAM Journal of Computing 32:754-792
6. Mixtacki J (2008) Genome Halving under DCJ Revisited Proceedings of COCOON'08, Springer, Lecture Notes in Computer Science 5092:276-286
7. Fertin G, Labarre A, Rusu I, Tannier E, Vialette S (2009) Combinatorics of Genome Rearrangements, MIT press
8. Gordon JL, Byrne KP, Wolfe KH (2009) Additions, Losses, and Rearrangements on the Evolutionary Route from a Reconstructed Ancestor to the Modern Saccharomyces cerevisiae Genome, PLoS Genetics 5(5): e1000485
9. Sankoff D (2009) Reconstructing the History of Yeast Genomes, PLoS Genet 5(5): e1000483
10. Seoighe C, Wolfe KH (1998) Extent of genomic rearrangement after genome duplication in yeast, Proc Natl Acad Sci U S A. 95(8):4447-52
11. Swenson KM, Arndt W, Tang J, Moret BME (2008) Phylogenetic reconstruction from complete gene orders of whole genomes, Proceedings of APBC'08, Imperial College Press, Advances in Bioinformatics and Computational Biology 6:241-250
12. Tannier E (2009) Yeast ancestral genome recontruction: the possibilities of automatic methods, Proceedings of RECOMB Comparative Genomics'09, Springer, Lecture Notes in Bioinformatics 5817:1-12
13. Tannier E, Zheng C, Sankoff D (2009) Multichromosomal Median and Halving Problems under Different Genomic Distances, BMC Bioinformatics 10:120
14. Warren R, Sankoff D (2009) Genome aliquoting with double cut and join, BMC Bioinformatics, 10(Suppl 1):S2
15. Xu AW (2008) A Fast and Exact Algorithm for the Median of Three Problem–A Graph Decomposition Approach, Proceedings of Recomb-Comparative Genomics'08, Springer, Lecture Notes in Bioinformatics 5267:184-197
16. Yancopoulos S, Attie O, Friedberg R (2005) Efficient sorting of genomic permutations by translocation, inversion and block interchange, Bioinformatics 21:3340-3346
17. Zhao H and Bourque G (2009) Recovering genome rearrangements in the mammalian phylogeny, Genome Research 19:934-942
18. Zheng C, Zhu Q, Sankoff D (2006) Genome halving with an outgroup, Evolutionary Bioinformatics 2:319-326
19. Zheng C, Zhu Q, Adam Z, Sankoff D (2008) Guided genome halving: hardness, heuristics and the history of the Hemiascomycetes, Bioinformatics 24(13):i96-104